

IN THE CLAIMS

This listing of claims will replace all prior versions and listings of claims in the Application:

LISTING OF CLAIMS:

1. (Currently Amended) A method of blocking attacks on a protected computer network, comprising:

receiving a plurality of packets from a network, each said packet having a packet time to live (TTL) value and belonging to a corresponding packet flow;

storing the smallest packet TTL value received from each said corresponding packet flow; and

prior to transmitting each said packet, setting said packet TTL value to said smallest packet TTL value received for said corresponding packet flow;

wherein storing the smallest packet TTL value received from each said corresponding packet flow includes, for each said packet:

if that packet is the first packet received from said corresponding packet flow, then storing the packet TTL value of that packet as said smallest packet TTL value received from said corresponding packet flow;

if that packet is not the first packet received from said corresponding packet flow and the packet TTL value of that packet is less than the stored smallest packet TTL value received from said corresponding packet flow, then storing the packet TTL value of that packet as said smallest packet TTL value received from said corresponding packet flow; and

if that packet is not the first packet received from said corresponding packet flow and the packet TTL value of that packet is greater than the stored smallest packet TTL value received from said corresponding packet flow, then refraining from storing the packet TTL value of that packet as said smallest packet TTL value received from said corresponding packet flow.

2. (Currently Amended) The method of Claim 1, wherein said storing the smallest packet TTL value further comprises:
 - associating an epoch with said stored smallest packet TTL value; and
 - if said epoch is greater than a predefined value, discarding said stored smallest packet TTL value.
3. (Original) The method of Claim 1, further comprising periodically resetting said stored smallest packet TTL value to a maximum value.
4. (Original) The method of Claim 1, wherein said setting said packet TTL value comprises:
 - determining if said corresponding packet flow is on an unrestricted list;
 - and if said corresponding packet flow is on said unrestricted list, setting said packet TTL value to a maximum value.
5. (Original) The method of Claim 1, wherein said setting said packet TTL value comprises:
 - determining if said corresponding packet flow is on an unrestricted list;
 - and if said corresponding packet flow is on said unrestricted list, leaving said packet TTL value unchanged.

6. (Currently Amended) An apparatus for blocking attacks on a protected computer network, comprising:

means for receiving a plurality of packets from a network, each said packet having a packet time to live (TTL) value and belonging to a corresponding packet flow;

means for storing the smallest packet TTL value received from each said corresponding packet flow; and

means for setting said packet TTL value to said smallest packet TTL value received for said corresponding packet flow prior to transmitting each said packet;

wherein said means for storing the smallest packet TTL value received from each said corresponding packet flow includes means for, for each said packet:

if that packet is the first packet received from said corresponding packet flow, then storing the packet TTL value of that packet as said smallest packet TTL value received from said corresponding packet flow;

if that packet is not the first packet received from said corresponding packet flow and the packet TTL value of that packet is less than the stored smallest packet TTL value received from said corresponding packet flow, then storing the packet TTL value of that packet as said smallest packet TTL value received from said corresponding packet flow; and

if that packet is not the first packet received from said corresponding packet flow and the packet TTL value of that packet is greater than the stored smallest packet TTL value received from said corresponding packet flow, then refraining from storing the packet TTL value of that packet as said smallest packet TTL value received from said corresponding packet flow.

7. (Currently Amended) The apparatus of Claim 6, wherein said means for storing the smallest packet TTL value further comprises:

means for associating an epoch with said stored smallest packet TTL value; and

means for discarding said stored smallest packet TTL value if said epoch is greater than a predefined value.

8. (Original) The apparatus of Claim 6, further comprising means for periodically resetting said stored smallest packet TTL value to a maximum value.

9. (Original) The apparatus of Claim 6, wherein said means for setting said packet TTL value comprises:

means for determining if said corresponding packet flow is on an unrestricted list; and

means for setting said packet TTL value to a maximum value if said corresponding packet flow is on said unrestricted list.

10. (Original) The apparatus of Claim 6, wherein said means for setting said packet TTL value comprises:

means for determining if said corresponding packet flow is on an unrestricted list; and

means for leaving said packet TTL value unchanged if said corresponding packet flow is on said unrestricted list.

11. (Currently Amended) An apparatus for blocking attacks on a protected computer network, comprising:

a packet classifier configured to receive a plurality of packets from a network, each said packet having a packet time to live (TTL) value and belonging to a corresponding packet flow;

a memory configured to store the smallest packet TTL value received from each said corresponding packet flow;~~and~~

a TTL rewrite unit configured to set said packet TTL value to said smallest packet TTL value received for said corresponding packet flow prior to transmitting each said packet; and

a controller, the controller being configured to, for each said packet:

if that packet is the first packet received from said corresponding packet flow, then store in memory the packet TTL value of that packet as said smallest packet TTL value received from said corresponding packet flow;

if that packet is not the first packet received from said corresponding packet flow and the packet TTL value of that packet is less than the stored smallest packet TTL value received from said corresponding packet flow, then store in memory the packet TTL value of that packet as said smallest packet TTL value received from said corresponding packet flow;
and

if that packet is not the first packet received from said corresponding packet flow and the packet TTL value of that packet is greater than the stored smallest packet TTL value received from said corresponding packet flow, then refrain from storing in memory the packet TTL value of that packet as said smallest packet TTL value received from said corresponding packet flow.

12. (Previously Presented) The apparatus of Claim 11, wherein said memory comprises:

first control means for associating an epoch with said stored smallest packet TTL value; and

second control means for discarding said stored smallest packet TTL value if said epoch is greater than a predefined value.

13. (Original) The apparatus of Claim 11, further comprising control means for periodically resetting said stored smallest packet TTL value to a maximum value.

14. (Original) The apparatus of Claim 11, wherein said TTL rewrite unit comprises:

first control means for determining if said corresponding packet flow is on an unrestricted list; and

second control means for setting said packet TTL value to a maximum value if said corresponding packet flow is on said unrestricted list.

15. (Original) The apparatus of Claim 11, wherein said TTL rewrite unit comprises:

first control means for determining if said corresponding packet flow is on an unrestricted list; and

second control means for leaving said packet TTL value unchanged if said corresponding packet flow is on said unrestricted list.

Claims 16-20 (Canceled).

21. (Currently Amended) A computer program product comprising a computer-readable medium having instructions stored thereon that, when performed by a computer, cause the computer to perform the following operations:

receiving a plurality of packets from a network, each said packet having a packet time to live (TTL) value and belonging to a corresponding packet flow;

storing the smallest packet TTL value received from each said corresponding packet flow; and

prior to transmitting each said packet, setting said packet TTL value to said smallest packet TTL value received for said corresponding packet flow;

wherein said instructions for storing the smallest packet TTL value received from each said corresponding packet flow comprise instructions that, when performed by the computer, cause the computer to perform the following operations:

if that packet is the first packet received from said corresponding packet flow, then storing the packet TTL value of that packet as said smallest packet TTL value received from said corresponding packet flow;

if that packet is not the first packet received from said corresponding packet flow and the packet TTL value of that packet is less than the stored smallest packet TTL value received from said corresponding packet flow, then storing the packet TTL value of that packet as said smallest packet TTL value received from said corresponding packet flow; and

if that packet is not the first packet received from said corresponding packet flow and the packet TTL value of that packet is greater than the stored smallest packet TTL value received from said corresponding packet flow, then refraining from storing the packet TTL value of that packet as said smallest packet TTL value received from said corresponding packet flow.

22. (Currently Amended) The computer program product of Claim 21, wherein said instructions for storing the smallest packet TTL value further comprise instructions that, when performed by the computer, cause the computer to perform the following operations:

associating an epoch with said stored smallest packet TTL value; and

if said epoch is greater than a predefined value, discarding said stored smallest packet TTL value.

23. (Previously Presented) The computer program product of Claim 21, further comprising instructions that, when performed by the computer, further cause the computer to perform the following operations:

periodically resetting said stored smallest packet TTL value to a maximum value.

24. (Previously Presented) The computer program product of Claim 21, wherein said instructions for setting said packet TTL value comprise instructions that, when performed by the computer, cause the computer to perform the following operations:

determining if said corresponding packet flow is on an unrestricted list;
and

if said corresponding packet flow is on said unrestricted list, setting said packet TTL value to a maximum value.

25. (Previously Presented) The computer program product of Claim 21, wherein said instructions for setting said packet TTL value comprise instructions that, when performed by the computer, cause the computer to perform the following operations:

determining if said corresponding packet flow is on an unrestricted list;
and

if said corresponding packet flow is on said unrestricted list, leaving said packet TTL value unchanged.

Claims 26-38 (Canceled).

39. (Previously Presented) The method of Claim 1, wherein:

-10-

for each said packet, said packet TTL value is a value stored within the header of that packet; and
the method further comprises transmitting each said packet across the protected computer network, said packet being configured to expire after a number of hops equal to said smallest packet TTL value received for said corresponding packet flow.